Detecting Aggressive Behavior in Discussion Threads Using Text Mining

Filippos Karolos Ventirozos[1], Iraklis Varlamis[1], and George Tsatsaronis[2]

[1] Department of Informatics and Telematics, Harokopio University of Athens, Greece,
{it21105,varlamis}@hua.gr,
WWW home page: https://www.dit.hua.gr/~varlamis/
[2] Content and Innovation, Elsevier B.V., Amsterdam, The Netherlands
g.tsatsaronis@elsevier.com

Abstract. The detection of aggressive behavior in online discussion communities is of great interest, due to the large number of users, especially of young age, who are frequently exposed to such behaviors in social networks. Research on cyberbullying prevention focuses on the detection of potentially harmful messages and the development of intelligent systems for the identification of verbal aggressiveness expressed with insults and threats. Text mining techniques are among the most promising tools used so far in the field of aggressive sentiments detection in short texts, such as comments, reviews, tweets etc. This article presents a novel approach which employs sentiment analysis at message level, but considers the whole communication thread (i.e., users discussions) as the context of the aggressive behavior. The suggested approach is able to detect aggressive, inappropriate or antisocial behavior, under the prism of the discussion context. Key aspects of the approach are the monitoring and analysis of the most recently published comments, and the application of text classification techniques for detecting whether an aggressive action actually emerges in a discussion thread. Thorough experimental validation of the suggested approach in a dataset for cyberbullying detection tasks demonstrates its applicability and advantages compared to other approaches.

Keywords: aggressive behavior, cyberbullying, sentiment analysis, thread classification

1 Introduction

Sentiment analysis methods aim at identifying the sentiment orientation of a piece of text (e.g., sentence, paragraph, snippet) by analyzing lexical features at word or term level. The problem is either handled as a binary classification problem [1] where only positive and negative sentiments are considered, or as a multi-class classification problem when a fine-grained list of sentiments is used (e.g., anger, disgust, fear, guilt, interest, joy, sadness, shame, surprise).

Despite the large number of works on sentiment analysis [2] and cyberbullying detection [3], text classification methods have focused only on single posts

and not yet on the complete discussion thread. Such methods have several disadvantages; for instance, they can be misled by attackers who intentionally misspell words to prevent detection [4], or they may falsely categorize the responses of the victims or their defenders as aggresive behavior. Existing methods actually neglect the fact that the inherent characteristics of bullying are *repetitiveness*, *intentionality* and *imbalance of power* between the harasser and the victim [5].

In an attempt to address these limitations and omissions we present in this work for the first time, to the best of our knowledge, a supervised learning model that detects aggressive behavior events by considering the whole thread in order to extract features which relate to changes in sentiment between consecutive messages. In order to validate experimentally the suggested approach we compared its performance in terms of accuracy in a benchmark set against a previously published state-of-the-art method which has been applied in the same set, and we also experimented with different variations of the method. The benchmark set is publicly available and comprises 139 discussion threads from *MySpace*. Results suggest that the presented method offers a more accurate predictor of aggressive behaviors in discussion threads.

The remainder of the article is organized as follows: Section 2 provides an overview of related literature. Section 3 summarizes the steps of the processing pipeline and highlights the novelties of the proposed methodology. Section 4 presents and discusses the experimental results, and, finally, Section 5 concludes and gives pointers to future work.

2 Related Work

The problem of textual harassment or aggressive behavior detection in text has been tackled by researchers as a classification problem. In [6], authors applied a supervised machine learning approach for detecting cyber-harassment, in which posts are represented using word frequency features, sentiment features and features that capture the similarity to neighboring posts. In [7] a rule-based model using a number of lexical features (e.g. bad words) outperformed the baseline bag-of-words (BOW) model. In [8] authors applied a range of binary and multi-class classifiers on a corpus of comments from YouTube videos in various topics. The findings show that topic-sensitive binary classifiers improved the performance of generic multi-class classifiers.

The authors in [9] compared an rule-based expert system, a supervised machine learning model, and a hybrid approach and showed that the latter outperformed the other two. In [10] a fuzzy support vector machine classifier using lexical features, sentiment features and user metadata was employed.

In [11] authors developed and applied a classification scheme for cyberbullying, which may detect cyberbullying presence, the judgment of its severity, and the role of the posts' authors (i.e., harasser, victim or bystander). Authors focused on specific cyberbullying-related text categories such as *threat/blackmail*, *insult*, and *curse/exclusion*, and the experimental results demonstrated the fea-

sibility of fine-grained cyberbullying detection. Character and word n-grams as well as lexicon based sentiment features were used.

All of the research works referenced so far approached the problem as a binary classification problem of single messages, without considering analyzing the entire thread. In addition, almost all of these approaches employed a very similar text pre-processing pipeline comprising stop-word removal, tokenization, POS tagging, emoticon detection, stemming, etc., and a typical text feature extraction step which resulted in bag-of-words, or, bag-of-stems representations that employ words, word and character n-grams, sentiment lexicon or even emoticon-related features, used for classifying texts at post level.

As a result, it is likely that these automated post labeling techniques may be inaccurate when an aggressive post does not contain bad words, when profanity or pronouns are misspelled, or when the posts are not in the language matching the aggressive words. In fact some works attempt to overcome these limitations by employing user-based features [10, 12], thus taking into account the history of users' activities. However, such features, or history, are not widely available, limiting from another angle this time, as opposed to the aforementioned approaches, their application at large scale and big heterogeneity of fora.

It is only recently that researchers focused on thread-level comment analysis to address such limitations. In [13], authors used thread-level features in a classification task, which exposed paid opinion manipulation trolls. One such feature was the number of times a certain users comments were among the top-k most loved/hated comments in some thread. However, the instances of the classification task were the users and not the thread messages. In [14], authors analyzed a question-answering community and used the whole communication thread as content, in a different application than the one discussed here, namely that of answer selection, and of evaluating the quality (good or bad) of given answers.

Perhaps closer to our work, in [15] authors focus on whole threads of comments. The authors acknowledge the fact that cyberbullying can take place even without the use of profane words. However, it is the occurrence of profane words in one or more comments in the same thread that is leveraged to decide whether cyberbullying is committed. The same authors in a more recent work [16], use an incremental classifier which sums the polarity of comments posted in a thread and decides when the thread must be blocked because of potential aggressiveness. Motivated by similar ideas, authors in [17], [18] and [19] agree that aggressive posts can be persistent and not single acts, thus highlight the need for whole thread processing.

The current work is distinguished, however, from the works in this latter category, which embed the notion of thread analysis, in the actual way the thread is used in the method. More precisely, in our work the thread is used to generate "sentiment n-grams" which represent the sequence of sentiments expressed within a thread, by the same, or different users that participate in the thread. We demonstrate experimentally that the consideration of this sequence reduces the effect of misclassifications at the comment-level and improves the performance of the aggressive behavior detection methodology overall.

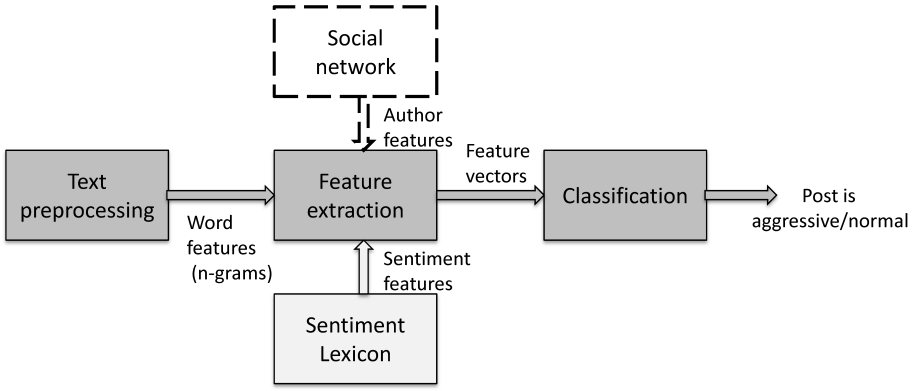

Fig. 1. Pipeline for detecting an aggressive behavior in text threads.

3 Cyber-bullying detection in text

3.1 Sentiment detection and cyberbullying

The two main approaches for extracting sentiment from text are lexicon- and machine learning-based. Lexicon-based approaches first calculate the semantic orientation of words or phrases in the text using one or more pre-compiled lexicons [20] (e.g., *SentiWordnet*[3], *Sentiful*[4], *ANEW*[5] *LIWC*[6], *WordNetAffect*[7], *SenticNet*[8]) and then decide on the document orientation and strength by adding up individual sentiment scores [21]. On the other side, machine learning methods build classifiers from labeled instances of texts or sentences and use a wide range of features in order to capture the orientation and strength of a sentiment in the text [22]. Support Vector Machine classifiers and Deep Learning approaches that use features such as word n-grams, with or without part-of-speech labels perform very well in this task [23].

The typical cyberbullying detection methodology in social media [24] as depicted in Figure 1. has two phases: First it extracts general keyword features, features for sentiments that are rare in other contexts but frequently expressed in bullying posts, and, possibly features that draw the author profile of each message. Then it classifies the message as aggressive or not.

3.2 The proposed method

The first step of the proposed methodology for aggressive behavior detection, as depicted in Figure 2, is the selection of a set of sentiments that will be used as

[3]http://sentiwordnet.isti.cnr.it/
[4]https://sites.google.com/site/alenaneviarouskaya/research-1/sentiful
[5]http://csea.phhp.ufl.edu/media/anewmessage.html
[6]http://liwc.wpengine.com/
[7]http://wndomains.fbk.eu/wnaffect.html
[8]http://sentic.net/downloads/

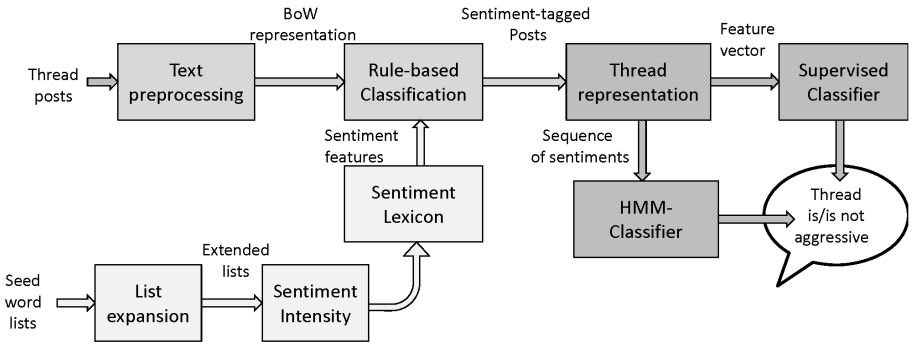

Fig. 2. The proposed pipeline for detecting an aggressive behavior in text threads.

features in the classification model. The proposed method considers 7 possible sentiments for a message: anger, disgust, fear, happiness, sadness, surprise and trust. These 7 have been characterised as basic emotions, clearly distinguished from other affective phenomena [25]. For these sentiments a lexicon is compiled, which contains related terms to each sentiment. Starting from a set of seed words for each sentiment, extracted from related sentiment lists on the web, we retrieve more synonym terms from *Wordnet* and expand the seed words to the final sentiment lexicon. The lexicon also contains a degree of strength for each sentiment word using information from ANEW, SentiWordnetand SenticNet 2 lexicons.

In the next step, using a rule based classifier[9] that takes into account the occurrence of sentiment words in the text, each message is tagged with one or more sentiments depending on our strategy, which will be explained in the following. The output of this step for a discussion thread is a sequence of sentiments, expressed by different users, which interchange during the discussion.

The last step comprises the thread classification algorithm, which decides whether an aggressive behavior is expressed within the thread.We evaluate different types of classifiers, that fall into two main categories depending on how the thread is represented: (i) when the thread is represented as a feature vector, then sentiment unigrams and bigrams are the features and any supervised classifier can be applied, and, (ii) when the thread is represented as an independent sequence of sentiments, then a Hidden Markov Model classifier is applied. The comparison of these two representations gives insight as to whether the order of the messages in the thread is important for the task.

[9] rule-based classifier, called *BullyTracer*, was used in [26] in the same dataset that we use in this work. However, any other classification method can be applied in this step.

4 Experimental evaluation

For the evaluation of the methodology, the (*Original*) dataset provided by the authors in [26] is used. The set comprises 139 discussion threads from *MySpace* forums, each containing 7 to 48 consecutive posts[10]. In the original work, authors acknowledge the interactive nature of cyberbullying, and process the conversations using a moving window of 10 posts to capture context (referenced as *window* dataset in the following). However, they never use sentiment n-grams as features as we do in our work. From the 139 threads, 39 discussions have been characterized for aggressive behavior (binary classification) and from the $2,062$ windows (each having 10 consecutive posts length), 425 have been marked for cyberbullying (binary classification), which creates an imbalance in the dataset. Our results are directly comparable to the original results. In the original results, accuracy ranges from 32% to 84%, the average overall accuracy is 58.63%, and the true positive ratio is 85.30%.

In our experiments, we evaluated the two representations and the classification alternatives presented in Figure 2 and compared our results against the original method and a random classifier. After pre-processing and rule-based classification, each post in the thread was tagged with the sentiments it contains, each one with a score, which corresponds to the total occurrences of the related sentiment words in the post.

4.1 Feature vector representation

In this alternative, each post is either classified to a single sentiment (the prevailing sentiment) or is tagged with multiple sentiment tags, depending on the sentiment words it contains. The post information is summed up at thread level and populates a feature vector that comprises as features:

- sentiment uni-grams ($Unig$), the degree of a sentiment in the thread (i.e., anger, disgust, fear, happiness, sadness, surprise, trust and neutral when no sentiment is expressed at all).
- sentiment bi-grams (Big), the occurrences of sentiment changes, among consecutive posts. The features are the 64 ordered sentiment combinations.
- personal pronouns(PP) e.g. I, me, you, him, it, they, etc., in consecutive posts, which frequently denote an aggressive stance towards another user.
- bullying bi-grams ($BBig$). When a post contains a word from the list of BullyTracer lexicon, then it is characterized as bullying (b) and neutral (n) in the opposite case. The feature counts the occurrences of different pairwise combinations (i.e. nn,nb,bn,bb) within the thread.
- bullying tri-grams ($BTrig$). The number of different triple combinations (nnn,nnb, etc) of bullying or neutral posts within the thread.

[10]The original dataset and the datasets we used in the current research can be downloaded from: https://goo.gl/wPrU2n

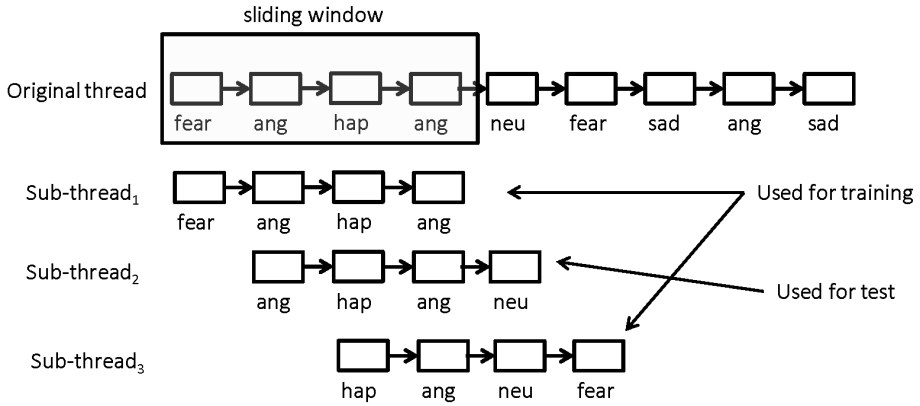

Fig. 3. An example of the dataset creation process and the possible similarities between training and test samples due to overlapping posts.

Sub-thread Classification We evaluated several supervised classification algorithms on different combinations of features using a 10-fold cross validation technique. All experiments were performed both in the *Original* and the *Window* dataset. Since the *Window* dataset was created using a sliding window over each thread it will not be fair to evaluate the algorithms using a completely random 10-fold split of the dataset, because in this case several highly overlapping subsequences of the same thread may split between the training and the test dataset as shown in Figure 3. It is indicative that a lazy 1-Nearest neighbor classifier achieved 95.43% accuracy, in such a type of split.

For this reason, in the *first set of experiments* we experimented with a 90%-10% training-test split of the dataset, taking care that no sub-threads of the same thread occur both in the training and test dataset. We repeated the experiment 10 times with an 40%-60% split of positive and negative samples in average in the test set and a respective split of 20%-80% in the training set. Using only combinations of uni-gram, bi-gram and personal pronoun features and a Radial Basis Function (RBF) classifier[11] an overall accuracy of 67.11% was achieved, whereas when an oversampling technique (*SMOTE*) was used to balance the number of positive an negative samples in the training dataset the overall accuracy dropped to 65.79%. Since the employed datasets are inbalanced, we also report the ROC area in each experiment in order to compare with a random classifier (ROC Area=0.5).

In the second experiment of this set, the same training-test splits were used but this time each post is tagged with a single sentiment. The performance of the *RBF* classifier dropped for most of the setups except for one that used the *SMOTE* over-sampled training dataset and all the features ($Unig + Big + PP$ feature set), which achieved an accuracy of 72.80%.

[11]The *RBFClassifier* implementation for *Weka* has been used.

Table 1. Sub-thread classification (*Window* dataset) using single and multi-class classification and sentiment uni-grams and bi-grams as features.

Features	Post class	Balanced Training	Accuracy	ROC Area
Unig	Multiple	No	60.96	0.529
Big	Multiple	No	61.40	0.532
Unig+Big	Multiple	No	61.84	0.569
Unig+Big	Multiple	Yes	64.91	0.595
Unig+Big+PP	Multiple	No	67.11	0.682
Unig+Big+PP	Multiple	Yes	65.79	0.621
Unig	Single	No	61.84	0.517
Big	Single	No	60.09	0.523
Unig+Big	Single	No	61.84	0.545
Unig+Big	Single	Yes	56.58	0.571
Unig+Big+PP	Single	No	63.60	0.754
Unig+Big+PP	Single	Yes	**72.81**	**0.777**

The results of this evaluation, which are summarized in Table 1 show that the combination of sentiment unigrams with bigrams and personal pronoun usage in consecutive posts can improve the overall performance. Also, the classification of each post to a single class (sentiment) in combination with the personal pronoun usage feature balances the impact of feature values to the final decision and achieves the best prediction performance so far.

Whole thread Classification In these experiments whole threads (*original* dataset) were classified in order to avoid the bias of fragmenting a conversation. The 10-fold cross validation strategy was directly applicable, since threads were not overlapping. The class distribution in the original dataset was 30%-70% with the majority being non-aggressive threads. In order to balance this ratio, a *SMOTE* filter was applied. Both types of post classification (single class and multi class) have been tested and the same feature set combinations have been evaluated (Uni-grams only, Bi-grams only, their combination and their combination plus the personal pronouns count). We also considered the user that posts each comment and merge any consecutive comments by the same user to a single comment. The only change in the results of this experiment is that the *RBF* classifier was outperformed by an *SVM* classifier[12] using a radial based kernel function. Results are depicted in Figure 2

Subthread Classification based on post type changes The last set of experiments that represented threads as feature vectors was based on the conversion of threads into a sequence of bullying or neutral posts. This binary classification of a post was performed using BullyTracer's lexicon and significantly reduces the number of features. More specifically when bullying bi-grams

[12]The *LibSVM* implementation of *Weka* with default parameters.

Table 2. Whole thread classification (*Original* dataset) using single and multi-class classification of posts and 10-fold cross validation.

Features	Post class	Balanced Training	Accuracy	ROC Area
Unig	Multiple	No	73.38	0.526
Big	Multiple	No	73.38	0.526
Unig+Big	Multiple	No	73.38	0.526
Unig+Big	Multiple	Yes	73.50	0.735
Unig+Big+PP	Multiple	No	74.45	0.542
Unig+Big+PP	Multiple	Yes	75.12	0.762
Unig	Single	No	73.38	0.500
Big	Single	No	71.94	0.500
Unig+Big	Single	No	72.66	0.521
Unig+Big	Single	Yes	72.50	0.725
Unig+Big+PP	Single	No	73.38	0.533
Unig+Big+PP	Single	Yes	**77.50**	**0.775**

are employed then we have only four features and in the case of tri-grams only eight features. We count the number of occurrences of each double or triple combination as well as the total number of bullying or non-bullying posts in the thread.

Experiments were performed on the *Window* dataset following the same test-training split as before. The algorithm that outperformed all others was the *RBF* classifier with a *PCA* attribute selection filter applied in a first step[13]. The results presented in Table 3 show an improved performance when compared to the respective results on the *Window* dataset with the sentiment features. They also show that using tri-grams instead of bi-grams in combination with simple counts of aggressive posts (both using BullyTracer lexicon and personal pronouns) gives a better performance. The above hold both when the training dataset is balanced or imbalanced, but the overall performance was better in the former case as expected.

4.2 Sequence representation

An alternative representation for a discussion thread is as an independent sequence of sentiments. In this case, only the prevailing sentiment is used for each post and a *Hidden Markov Model* classifier is applied. Since the *SMOTE* over-sampling method cannot be applied in the sequence attribute, we apply a down-sampling technique (remove majority class samples without replacement) in order to balance the training dataset. From the results in Table 4 we see that the *HMM* classifier does not perform well with the sentiment sequences, probably because a larger training set is needed. The performance of the classifier for the sequence of aggressive/neutral posts is still high and comparable to the

[13] AttributeSelectedClassifier with *PCA* as attribute selection method and *RBF* classifier as classification method, was used.

Table 3. Sub-thread classification threads using single-class classification of posts, a training-test split and Bullying/Neutral bi-grams and tri-grams as features.

Features	Balanced Training	Accuracy	ROC Area
Bcount	No	69.74	0.808
Bcount	Yes	75.00	0.808
Bbig+Bcount	No	67.98	0.803
Bbig+Bcount	Yes	72.81	0.801
Bbig+Bcount+PP	No	71.05	0.879
Bbig+Bcount+PP	Yes	76.75	0.871
Btrig+Bcount	No	69.74	0.797
Btrig+Bcount	Yes	76.32	0.817
Btrig+Bcount+PP	No	69.30	0.870
Btrig+Bcount+PP	Yes	**79.39**	**0.867**

Table 4. Classification of threads and sub-threads as sequences of sentiments or aggressive/neutral posts. An HMM classifier was used in all cases.

Features	Split	Dataset	Balanced Training	Accuracy	ROC Area	TP ratio
Sentiments	90-10	Window	No	51.32	0.532	**94.6**
Sentiments	90-10	Window	Yes	53.07	0.517	79.3
Sentiments	10 fold	Original	No	49.64	0.457	48.7
Sentiments	10 fold	Original	Yes	56.41	0.504	61.5

classifiers that use feature representations. Although the accuracy scores are not high, the reported true positive rate in the *Window* dataset, when the original data without sampling are employed is really high (94.6%), higher than that report in the original work.

5 Conclusions

This article presents a novel approach for the detection of aggressive and antisocial behavior in discussion threads, using text mining. The proposed method processes the thread of messages as a whole and captures the changes in sentiment between consecutive posts, which are used in turn for classifying the whole thread as aggressive or neutral. This reduces the effect of misclassifications in message-level and improves the performance of the aggressive behavior detection methodology. A set of n-gram like features, that capture the change of sentiment in consecutive posts or the interchange between bullying and neutral posts, as well as the use of personal pronouns in consecutive posts, are combined and evaluated. Experimental evaluation on a publicly available dataset shows that the proposed method outperforms a related, state-of-the-art method applied in the same dataset. As our future work, we plan to investigate the representation of sentiments as word embeddings directly learned from deep neural network architectures, such as *long short-term memory recurrent neural networks*.

References

1. Katakis, I.M., Varlamis, I., Tsatsaronis, G.: PYTHIA: Employing lexical and semantic features for sentiment analysis. In: Proc. of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD), Springer (2014) 448–451
2. Medhat, W., Hassan, A., Korashy, H.: Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal **5**(4) (2014) 1093–1113
3. Van Royen, K., Poels, K., Daelemans, W., Vandebosch, H.: Automatic monitoring of cyberbullying on social networking sites: From technological feasibility to desirability. Telematics and Informatics **32**(1) (2015) 89–97
4. Moore, M.J., Nakano, T., Enomoto, A., Suda, T.: Anonymity and roles associated with aggressive posts in an online forum. Computers in Human Behavior **28**(3) (2012) 861–867
5. Olweus, D.: Bullying at school: Knowledge base and an effective intervention programa. Annals of the New York Academy of Sciences **794**(1) (1996) 265–276
6. Yin, D., Xue, Z., Hong, L., Davison, B.D., Kontostathis, A., Edwards, L.: Detection of harassment on web 2.0. In: Proc. of Content Analysis in the WEB 2.0 (CAW2.0) Workshop at WWW. (2009)
7. Reynolds, K., Kontostathis, A., Edwards, L.: Using machine learning to detect cyberbullying. In: Proc. of Machine Learning and Applications and Workshop (ICMLA). Volume 2., IEEE (2011) 241–244
8. Dinakar, K., Reichart, R., Lieberman, H.: Modeling the detection of textual cyberbullying. In: Proc. of the Workshop on the Social Mobile Web, at the International AAAI Conference on Weblogs and Social Media. (2011)
9. Dadvar, M., Trieschnigg, D., de Jong, F.: Experts and machines against bullies: A hybrid approach to detect cyberbullies. In: Proc. of the Canadian Conference on Artificial Intelligence, Springer (2014) 275–281
10. Nahar, V., Al-Maskari, S., Li, X., Pang, C.: Semi-supervised learning for cyberbullying detection in social networks. In: Proc. of the4 25th Australasian Database Conference (ADC), Springer (2014) 160–171
11. Van Hee, C., Lefever, E., Verhoeven, B., Mennes, J., Desmet, B., De Pauw, G., Daelemans, W., Hoste, V.: Detection and fine-grained classification of cyberbullying events. In: Proc. of the Recent Advances in NLP Conference (RANLP). (2015) 672–680
12. Dadvar, M., de Jong, F., Ordelman, R., Trieschnigg, D.: Improved cyberbullying detection using gender information. In: Proc. of the Dutch-Belgian Information Retrieval Workshop (DIR). (2012) 23–25
13. Mihaylov, T., Nakov, P.: Hunting for troll comments in news community forums. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics (ACL). Volume 16. (2016) 399–405
14. Joty, S., Barrón-Cedeno, A., Da San Martino, G., Filice, S., Marquez, L., Moschitti, A., Nakov, P.: Global thread-level inference for comment classification in community question answering. In: Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP). Volume 15. (2015)
15. Rafiq, R.I., Han, H.H.R., Lv, Q., Mishra, S., Mattso, S.A.: Careful what you share in six seconds: Detecting cyberbullying instances in vine. In: Proc. of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE (2015) 617–622

16. Rafiq, R.I., Hosseinmardi, H., Han, R., Lv, Q., Mishra, S.: Investigating factors influencing the latency of cyberbullying detection. arXiv preprint:1611.05419 (2016)
17. Smith, P.K., Mahdavi, J., Carvalho, M., Fisher, S., Russell, S., Tippett, N.: Cyberbullying: Its nature and impact in secondary school pupils. Journal of child psychology and psychiatry **49**(4) (2008) 376–385
18. Dooley, J.J., Pyżalski, J., Cross, D.: Cyberbullying versus face-to-face bullying: A theoretical and conceptual review. Journal of Psychology **217**(4) (2009) 182–188
19. Grigg, D.W.: Cyber-aggression: Definition and concept of cyberbullying. Australian Journal of Guidance and Counselling **20**(02) (2010) 143–156
20. Poria, S., Gelbukh, A., Cambria, E., Yang, P., Hussain, A., Durrani, T.: Merging senticnet and wordnet-affect emotion lists for sentiment analysis. In: Proc. of the 11th International Conference on Signal Processing (ICSP). Volume 2., IEEE (2012) 1251–1255
21. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Computational linguistics **37**(2) (2011) 267–307
22. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology **61**(12) (2010) 2544–2558
23. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: A deep learning approach. In: Proc. of the 28th International Conference on Machine Learning (ICML). (2011) 513–520
24. Xu, J.M., Jun, K.S., Zhu, X., Bellmore, A.: Learning from bullying traces in social media. In: Proc. of the North American Chapter of the Association for Computational Linguistics (NAACL), Association for Computational Linguistics (2012) 656–666
25. Ekman, P.: An argument for basic emotions. Cognition & emotion **6**(3-4) (1992) 169–200
26. Bayzick, J., Kontostathis, A., Edwards, L.: Detecting the presence of cyberbullying using computer software. In: Proc. of the 3rd International Conference on Web Science (WebSci). (2011)